MicroTCA-based LLRF Control with EPICS

Patrick Nonn

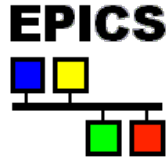
Berlin, 07.09.2022

Motivation

- The MicroTCA-based LLRF control system at the XFEL uses the, in-house developed, **D**istributed **O**bject-**O**riented **C**ontrol **S**ystem (DOOCS)
- In order to grow the MicroTCA ecosystem,
  we want to promote MicroTCA outside of DESY
- As DOOCS is not used outside of DESY,
  ChimeraTK/Control System Adapter was developed
- EPICS is widely used at various scientific facilities, especially particle accelerator labs

Introduction to EPICS - Background

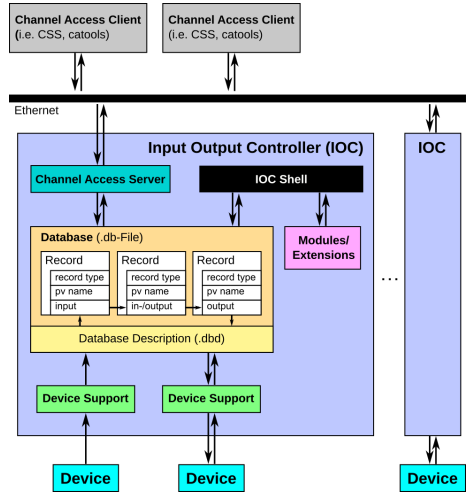
- Initially developed as Ground Test Accelerator Control System (GTACS) in 1988 at Los Alamos National Laboatory
- In a cooperation between Los Alamos and Argonne National Laboratories, GTACS was expended and renamed EPICS
- First presented at ICALEPCS in 1991
- Used todayat over 50 large science facilities (particle accelerators, observatories and nuclear fusion installations) including DESY and BESSY



Experimental
Physics and
Industrial
Control
System

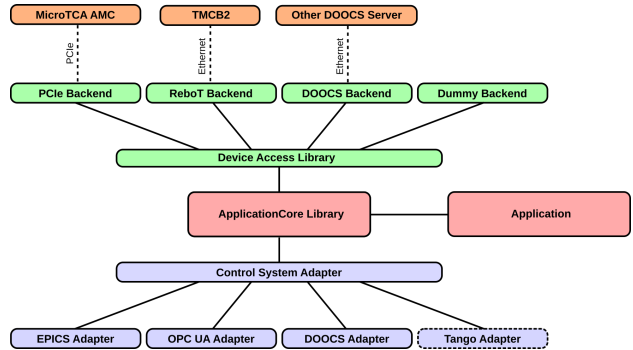
Introduction to EPICS - Concept

- Distributed Control System
- Database of Process Variables (PVs)
- PVs are accessed remotely via either Channel Access (EPICS 3) or pvAccess (EPICS 7) protocol
- Support for Channel Access / pvAccess available in a wide range of software (MATLAB, Python, Control System Studio, Command Line Tools,…)
- Device Support Modules allow to access devices/buses



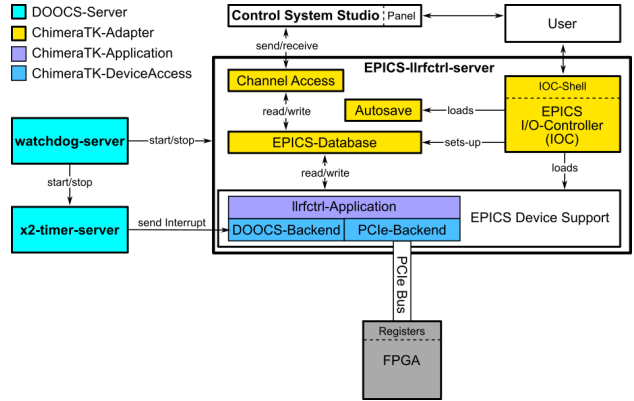
ChimeraTK Overview

- ChimeraTK allows to develop applications, independent from the control system
- An adapter for EPICS 3.16 exists, one for EPICS 7 is under development
- Available on Github https://github.com/ChimeraTK



EPICS-based LLRF Control Server

- ChimeraTK-Application and -DeviceAccess act as an EPICS Device Support, which can be accessed by the database
- Some DOOCS servers (Watchdog, x2timer) are still needed
- The EPICS-Database has to be provided by the server configuration



EPICS-based Server Configuration - Mapping

- Firmware provides a mapp-file to map register addresses to register names
- The ChimeraTK-application can require xlmap-files
- The server configuration has to provide a dmap file, assigning mapping files to devices

*.mapp-file

```
@MAPFILE_REVISION 1.1.4-2-g078e9dfc
BSP.WORD_ID             1  0 4 0 32 0 0 RO
BSP.WORD_VERSION        1  4 4 0 32 0 0 RO
BSP.WORD_PRJ_ID         1  8 4 0 32 0 0 RO
BSP.WORD_PRJ_VERSION    1 12 4 0 32 0 0 RO
BSP.WORD_PRJ_SHASUM     1 16 4 0 32 0 0 RO
...
```

*.xlmap-file

```
<logicalNameMap>
  <module name="Controller">
    <module name="SetPoint">
      <module name="Table">
        <redirectedRegister name="Q">
          <targetDevice>CtrlBoard</targetDevice>
          <targetRegister>CTABLES.AREA_SP_Q</targetRegister>
        </redirectedRegister>
      </module>
      <module name="DAQ">
        <redirectedChannel name="Q">
          <targetDevice>CtrlBoard</targetDevice>
          <targetRegister>APP.DAQ0_BUF0</targetRegister>
          <targetChannel>3</targetChannel>
          <targetStartIndex>0</targetStartIndex>
          <numberOfElements>2048</numberOfElements>
        </redirectedChannel>
      </module>
    </module> <!-- SetPoint -->
  </module> <!-- Controller -->
</logicalNameMap>
```

*.dmap-file

```
x2timer     (doocs:TEST.DOOCS/X2TIMER/NEUROMANCER.0?cacheFile=timer.cache)
Timer       (logicalNameMap?map=timer.xlmap)
CtrlBoard   (pci:pcieunis3?map=./llrf_scav_sis8300ku_regae_1.1.4-2-g078e9dfc.mapp)
Controller  (logicalNameMap?map=llrfctrl_controller.xlmap)
```

EPICS-based Server Configuration- EPICS Database

- ChimeraTK provides a tool to extract all the variables from the application and dump it into an xml-file
- These Variables are organized in a tree-structure and need to be mapped to a “flat” EPICS database
- The dbGenerator tool is available on Github, to generate EPICS databases from variable files and a configuration file.

Variable xml-file

```
<application xmlns="https://github…" name="llrfctrl">
  <directory name="Controller">
    <directory name="Status">
      <variable name="overall">
        <value_type>int32</value_type>
        <direction>application_to_control_system</direction>
        <unit></unit>
        <description>Controller configuration…</description>
        <numberOfElements>1</numberOfElements>
      </variable>
    </directory>
  </directory>
</application>
```

EPICS Database Configurationn File

```
<EPICSdb xmlns="<namespace>" application="llrfctrl">
  <sourcefile type="xml-variables" path="llrfctrl.xml" label="llrfctrl">
    <alias handle="Ctrl" surrogate="Controller/" />
    <alias handle="CtrlFf" surrogate="+{Ctrl}+{Ff}" />
    <alias handle="Ff" surrogate="FeedForward/" />
  </sourcefile>
  <outputfile path="<path>" autosavePath="<path>" macroReserve="8">
    <field type="DTYP" value="ChimeraTK" />
    <recordgroup type="aai" autosave="false">
      <field type="SCAN" value="1 second" />
      <field type="INP" value="@${APP) +{:address}" />
      <field type="EGU" value="+{:unit}" />
      <field type="FTVL" value="+{:value_type}" />
      <field type="NELM" value="+{:numberOfElements}" />
      <field type="PREC" value="6" />
      <record pvName="${Server)/CtrlFF/FTRatio" source="llrfctrl.+{CtrlFf}flattopRatio" />
      <record pvName="${Server)/B0Cfg/BCF" source="llrfctrl.+{Adcb0Cfg}BCF">
        <field type="EGU" value="MHz" />
      </record>
    </recordgroup>
  </outputfile>
```

EPICS-based LLRF Control System at HZB

- EPICS-based LLRF control system sucessfully implemented at BESSY and Sealab
- Debian package for epics-llrfctrl-server available at public repository of DESY (http://doocs.desy.de/pub/doocs)
- Panels for the llrfctrl-server CS-Studio Phoebus available (from me)

Things to come: Generic Server

- llrfctrl-server only works with the matching firmware
- It is possible to “publish” the registers as PVs with the generic server, either to implement some logic directly in the EPICS database, or to develop some distributed high-level application, using i.e. pyepics
- Basically running, but some bugfixing required

Summary

- DESY/MSK promotes/supports EPICS-based LLRF control systems
- EPICS R7 support is coming soon
- EPICS-based llrfctrl-server is publicly available
- EPICS-based generic server is under development

Thank you!

Contact

DESY. Deutsches Elektronen-Synchrotron

www.desy.de

Patrick Nonn
0000-0003-4492-9420
MSK
patrick.nonn@desy.de
+49–40–8998–1962